JAVA.UTIL.IDENTITYHASHMAP CLASS

Introduction

The **java.util.IdentityHashMap** class imlements the Map interface with a hash table, using reference-equality in place of object-equality when comparing keys (and values).Following are the important points about IdentityHashMap:

- This class provides all of the optional map operations, and permits null values and the null key.

- This class makes no guarantees as to the order of the map; in particular, it does not guarantee that the order will remain constant over time.

- In an IdentityHashMap, two keys k1 and k2 are considered equal if and only if (k1==k2), while in Map implementations (like HashMap) two keys k1 and k2 are considered equal if and only if (k1==null ? k2==null : k1.equals(k2)).

Class declaration

Following is the declaration for **java.util.IdentityHashMap** class:

```
public class IdentityHashMap<K,V>
   extends AbstractMap<K,V>
       implements Map<K,V>, Serializable, Cloneable
```

Class constructors

S.N.	Constructor & Description
1	**IdentityHashMap()** This constructs a new, empty identity hash map with a default expected maximum size (21).
2	**IdentityHashMap(int expectedMaxSize)** This constructs a new, empty map with the specified expected maximum size.
3	**IdentityHashMap(Map<? extends K,? extends V> m)** This constructs a new identity hash map containing the keys-value mappings in the specified map.

Class methods

S.N.	Method & Description
1	void clear() This method removes all of the mappings from this map.
2	Object clone() This method returns a shallow copy of this identity hash map: the keys and values themselves are not cloned.
3	boolean containsKey(Object key) This method tests whether the specified object reference is a key in this identity hash map.

4	**boolean containsValue(Object value)** This method tests whether the specified object reference is a value in this identity hash map.
5	**Set<Map.Entry<K,V>> entrySet()** This method returns a Set view of the mappings contained in this map.
6	**boolean equals(Object o)** This method compares the specified object with this map for equality.
7	**V get(Object key)** This method returns the value to which the specified key is mapped, or null if this map contains no mapping for the key.
8	**int hashCode()** This method returns returns the hash code value for this map.
9	**boolean isEmpty()** This method returns true if this identity hash map contains no key-value mappings.
10	**Set<K>keySet()** This method returns an identity-based set view of the keys contained in this map.
11	**V put(K key, V value)** This method associates the specified value with the specified key in this identity hash map.
12	**void putAll(Map<? extends K,? extends V> m)** This method copies all of the mappings from the specified map to this map.
13	**V remove(Object key)** This method removes the mapping for this key from this map if present.
14	**int size()** This method returns the number of key-value mappings in this identity hash map.
15	**Collection<V> values()** This method returns returns a Collection view of the values contained in this map.

Methods inherited

This class inherits methods from the following classes:

- java.util.AbstractMap

- java.util.Object